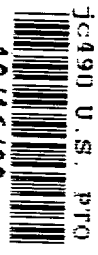


12-20-99

A

12/16/99



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Inventorship.....Omoigui
Applicant..... Microsoft Corporation
Attorney's Docket No. MS1-362US
Title: Searching and Recording Media Streams



TRANSMITTAL LETTER AND CERTIFICATE OF MAILING

To: Commissioner of Patents and Trademarks
Washington, D.C. 20231
From: Allan T. Sponseller (509) 324-9256
Lee & Hayes, PLLC
421 W. Riverside Avenue, Suite 500
Spokane, WA 99201

The following enumerated items accompany this transmittal letter and are being submitted for the matter identified in the above caption.

- 1. Transmittal Letter with Certificate of Mailing included.
2. PTO Return Postcard Receipt
3. Fee Transmittal
4. New patent application (title page plus 36 pages, including claims 1-43 & Abstract)
5. Executed Declaration
6. 8 sheets of formal drawings (Figs. 1-8)
7. Assignment w/Recordation Cover Sheet

Large Entity Status [x] Small Entity Status []

The Commissioner is hereby authorized to charge payment of fees or credit overpayments to Deposit Account No. 50-0463 in connection with any patent application filing fees under 37 CFR 1.16, and any processing fees under 37 CFR 1.17.

Date: 12/16/99
By: [Signature]
Allan T. Sponseller
Reg. No. 38,318

CERTIFICATE OF MAILING

I hereby certify that the items listed above as enclosed are being deposited with the U.S. Postal Service as either first class mail, or Express Mail if the blank for Express Mail No. is completed below, in an envelope addressed to The Commissioner of Patents and Trademarks, Washington, D.C. 20231, on the below-indicated date. Any Express Mail No. has also been marked on the listed items.

Express Mail No. (if applicable) EL472378552

Date: 12/16/1999
By: [Signature]
Lori A. Vierra

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR LETTERS PATENT

Searching and Recording Media Streams

Inventor(s):
Nosakhare D. Omoigui

ATTORNEY'S DOCKET NO. MS1-362US

TECHNICAL FIELD

This invention relates to networked client/server systems and to searching and recording streaming media content in such systems.

BACKGROUND OF THE INVENTION

Multimedia streaming—the continuous delivery of synchronized media data like video, audio, text, and animation—is a critical link in the digital multimedia revolution. Today, streaming media is primarily about video and audio, but a richer, broader digital media era is emerging with a profound and growing impact on the Internet and digital broadcasting.

Synchronized media means multiple media objects that share a common timeline. Video and audio are examples of synchronized media—each is a separate data stream with its own data structure, but the two data streams are played back in synchronization with each other. Virtually any media type can have a timeline. For example, an image object can change like an animated .gif file, text can change and move, and animation and digital effects happen over time. This concept of synchronizing multiple media types is gaining greater meaning and currency with the emergence of more sophisticated media composition frameworks implied by MPEG-4, Dynamic HTML, and other media playback environments.

The term “streaming” is used to indicate that the data representing the various media types is provided over a network to a client computer on a real-time, as-needed basis, rather than being pre-delivered in its entirety before playback. Thus, the client computer renders streaming data as it is received from a network server, rather than waiting for an entire “file” to be delivered.

1 Streaming multimedia content enables a variety of informational content
2 that was not previously available over the Internet or other computer networks.
3 Live content is one significant example of such content. Using streaming
4 multimedia, audio, video, or audio/visual coverage of noteworthy events can be
5 broadcast over the Internet as the events unfold. Similarly, television and radio
6 stations can transmit their live content over the Internet.

7 However, one current problem with streaming multimedia content is that
8 users are typically limited to accessing the multimedia content via common
9 "shuttle controls" on a multimedia player, such as a play button, fast forward
10 button, pause button, etc. Given that large amounts of data can be stored as
11 multimedia content (e.g., individual presentations lasting for hours), such controls
12 make it difficult for a user to locate the portions of the multimedia content that are
13 of most interest to him or her.

14 An additional problem with streaming multimedia content is that the user
15 must typically be connected to the same network as the server (e.g., the Internet)
16 in order to receive the streaming multimedia content. If this connection is not
17 maintained then the streaming of the multimedia content stops. This "continuous
18 connection" limitation can be troublesome for many individuals, such as those
19 using portable computers in locations that may not always have access to the
20 appropriate network, or individuals who do not want to tie up a telephone line for
21 their network connection while playing back the multimedia content.

22 The invention described below addresses these disadvantages, providing for
23 the searching and recording of streaming media content.
24
25

SUMMARY OF THE INVENTION

In a networked client/server system, media content is streamed from the server to the client. A user of the client can search the media content to identify temporal locations that satisfy certain search criteria, and/or store the media content locally at the client for subsequent playback.

According to one aspect of the invention, indexes are maintained for each of different media streams that can be streamed to the client either individually or together for a multimedia presentation. The indexes store a correspondence between content for a media stream and temporal locations of that media stream. In response to a user search request, search criteria is compared to the appropriate index(es) to identify whether the search criteria matches any data in the index(es).

According to another aspect of the invention, in response to a user search request the search criteria from the search request is compared directly to the media stream data rather than to an index. This comparison is made to identify whether the search criteria matches any of the media stream data.

According to another aspect of the invention, if data matching the search criteria is found (either in an index or the media stream data), then the media server “seeks” to a temporal location of the media stream identified by the matching data. The server then proceeds to stream the media content to the client beginning at that temporal location.

According to another aspect of the invention, a search request and corresponding search criteria are compared to multiple media streams (either directly or indirectly via associated indexes). Thus, a single search request can be used to search through all of the media streams of a multimedia presentation.

1 According to another aspect of the invention, the multiple media streams of
2 a multimedia presentation are streamed from the server to the client and stored
3 locally by the client. A markup document, referencing the multiple media streams
4 also stored locally at the client, is generated and stored at the client. Thus, a user
5 can play back the locally stored multimedia presentation at a later time when not
6 coupled to a network and thus not able to receive streaming media from the server.

8 **BRIEF DESCRIPTION OF THE DRAWINGS**

9 The present invention is illustrated by way of example and not limitation in
10 the figures of the accompanying drawings. The same numbers are used
11 throughout the figures to reference like components and/or features.

12 Fig. 1 shows a client/server network system and environment in accordance
13 with the invention.

14 Fig. 2 shows a general example of a computer that can be used in
15 accordance with the invention.

16 Fig. 3 illustrates an exemplary client-server relationship for streaming data.

17 Fig. 4 is a flowchart illustrating an exemplary process for searching media
18 streams in accordance with one implementation of the invention.

19 Fig. 5 is a flowchart illustrating another exemplary process for searching
20 media streams in accordance with another implementation of the invention.

21 Figs. 6 and 7 are block diagrams illustrating the local storage of a
22 multimedia presentation in accordance with one implementation of the invention.

23 Fig. 8 is a flowchart illustrating an exemplary process for recording a
24 multimedia presentation in accordance with one implementation of the invention.
25

DETAILED DESCRIPTION

General Network Structure

Fig. 1 shows a client/server network system and environment in accordance with the invention. Generally, the system includes one or more (m) network multimedia server computers 100, one or more (z) index server computers 102, and one or more (n) network client computers 104. The computers communicate with each other over a data communications network, which in Fig. 1 includes a public network 106 such as the Internet. The data communications network might also include local-area networks and/or private wide-area networks. Server computers 100 and client computers 104 communicate with one another via any of a wide variety of known protocols, such as the Hypertext Transfer Protocol (HTTP).

Multimedia servers 100 have access to streaming media content in the form of different media streams. These media streams can be individual media streams (e.g., audio, video, graphical, text, etc.), or alternatively composite media streams including multiple such individual streams. Some media streams might be stored as files 108 in a database or other file storage system, while other media streams 110 might be supplied to the server on a “live” basis from other data source components through dedicated communications channels or through the Internet itself.

The media streams received from servers 100 are rendered at the client computers 104 as a multimedia presentation, which can include media streams from one or more of the servers 100. These different media streams can include one or more of the same or different types of media streams. For example, a

1 multimedia presentation may include two video streams, one audio stream, and
2 one stream of graphical images.

3 A user interface (UI) at the client computer 104 allows users to control the
4 playback of the multimedia presentation, such as selecting which of multiple
5 presentations to play back, controlling pausing of the playback, etc. The UI at
6 client 104 further allows a user to input search criteria for searching one or more
7 of the individual media streams available from a server 100, and to save the media
8 streams of a multimedia presentation for subsequent playback when not coupled to
9 network 106.

10 Index servers 102 optionally maintain indexes for the streaming media data
11 available from servers 100. These indexes provide a correspondence between
12 elements or objects of the media data streams and temporal locations of the media
13 data streams. These indexes can be used for searching the media data streams, as
14 discussed in more detail below. Alternatively, the indexes may be maintained at
15 the media servers 100.

16 17 **Streaming Media**

18 In this discussion, streaming media refers to one or more individual media
19 streams being transferred over a network to a client computer on an as-needed
20 basis rather than being pre-delivered in their entirety before playback. Each of the
21 individual media streams corresponds to and represents a different media type and
22 each of the media streams can be rendered by a network client to produce a user-
23 perceivable presentation using a particular presentation medium. The individual
24 media streams can be rendered to produce a plurality of different types of user-
25 perceivable media, including synchronized audio or sound, video graphics or

1 motion pictures, animation, textual content, command script sequences, or other
2 media types that convey time-varying information or content in a way that can be
3 sensed and perceived by a human. The individual media streams have their own
4 timelines, which are synchronized with each other so that the media streams can
5 be rendered simultaneously for a coordinated multimedia presentation. These
6 individual media streams can be delivered to the client computer as individual
7 streams from one or more servers, as a composite media stream(s) from one or
8 more servers, or a combination thereof.

9 In this discussion, the term “composite media stream” describes
10 synchronized streaming data that represents a segment of multimedia content. The
11 composite media stream has a timeline that establishes the speed at which the
12 content is rendered. The composite media stream can be rendered to produce a
13 plurality of different types of user-perceivable media, such as synchronized audio
14 or sound, video graphics or motion pictures, animation, textual content, command
15 script sequences, etc. A composite media stream includes a plurality of individual
16 media streams representing the multimedia content.

17 There are various standards for streaming media content and composite
18 media streams. “Advanced Streaming Format” (ASF) is an example of such a
19 standard, including both accepted versions of the standard and proposed standards
20 for future adoption. ASF specifies the way in which multimedia content is stored,
21 streamed, and presented by the tools, servers, and clients of various multimedia
22 vendors. ASF provides benefits such as local and network playback, extensible
23 media types, component download, scalable media types, prioritization of streams,
24 multiple language support, environment independence, rich inter-stream
25

relationships, and expandability. Further details about ASF are available from Microsoft Corporation of Redmond, Washington.

Regardless of the streaming format used, an individual data stream contains a sequence of digital data units that are rendered individually, in sequence, to produce an image, sound, or some other stimuli that is perceived by a human to be continuously varying. For example, an audio data stream comprises a sequence of sample values that are converted to a pitch and volume to produce continuously varying sound. A video data stream comprises a sequence of digitally-specified graphics frames that are rendered in sequence to produce a moving picture. An animation stream comprises a sequence of graphical images that are rendered in sequence to produce a moving image. An image stream comprises a sequence of graphical images that are rendered to produce a changing image over time. A text stream is a sequence of symbols and/or alphanumeric characters that are rendered to produce different symbol/character combinations over time (e.g., in the form of words).

For a composite media stream, the individual data streams are typically interleaved in a single sequence of data packets. Various types of data compression might be used within a particular data format to reduce communications bandwidth requirements.

The sequential data units (such as audio sample values, video frames, groups of characters, graphical images, etc.) of the individual streams are associated with both delivery times and presentation times, relative to an arbitrary start time. The delivery time of a data unit indicates when the data unit should be delivered to a rendering client. The presentation time indicates when the value

1 should be actually rendered. Normally, the delivery time of a data unit precedes
2 the presentation time.

3 The presentation times determine the actual speed of playback. For data
4 streams representing actual events or performances, the presentation times
5 correspond to the relative times at which the data samples were actually recorded.
6 The presentation times of the various different individual data streams are
7 consistent with each other so that the streams remain coordinated and
8 synchronized during playback.

9 10 **Exemplary Computer Environment**

11 In the discussion below, the invention will be described in the general
12 context of computer-executable instructions, such as program modules, being
13 executed by one or more conventional personal computers. Generally, program
14 modules include routines, programs, objects, components, data structures, etc. that
15 perform particular tasks or implement particular abstract data types. Moreover,
16 those skilled in the art will appreciate that the invention may be practiced with
17 other computer system configurations, including hand-held devices,
18 multiprocessor systems, microprocessor-based or programmable consumer
19 electronics, network PCs, minicomputers, mainframe computers, and the like. In a
20 distributed computer environment, program modules may be located in both local
21 and remote memory storage devices.

22 Alternatively, the invention could be implemented in hardware or a
23 combination of hardware, software, and/or firmware. For example, one or more
24 application specific integrated circuits (ASICs) could be programmed to carry out
25 the invention.

Fig. 2 shows a general example of a computer 142 that can be used in accordance with the invention. Computer 142 is shown as an example of a computer that can perform the functions of any of server computers 100 or 102, or client computers 104 of Fig. 1.

Computer 142 includes one or more processors or processing units 144, a system memory 146, and a bus 148 that couples various system components including the system memory 146 to processors 144.

The bus 148 represents one or more of any of several types of bus structures, including a memory bus or memory controller, a peripheral bus, an accelerated graphics port, and a processor or local bus using any of a variety of bus architectures. The system memory includes read only memory (ROM) 150 and random access memory (RAM) 152. A basic input/output system (BIOS) 154, containing the basic routines that help to transfer information between elements within computer 142, such as during start-up, is stored in ROM 150. Computer 142 further includes a hard disk drive 156 for reading from and writing to a hard disk, not shown, a magnetic disk drive 158 for reading from and writing to a removable magnetic disk 160, and an optical disk drive 162 for reading from or writing to a removable optical disk 164 such as a CD ROM or other optical media. The hard disk drive 156, magnetic disk drive 158, and optical disk drive 162 are connected to the system bus 148 by an SCSI interface 166 or some other appropriate interface. The drives and their associated computer-readable media provide nonvolatile storage of computer readable instructions, data structures, program modules and other data for computer 142. Although the exemplary environment described herein employs a hard disk, a removable magnetic disk 160 and a removable optical disk 164, it should be appreciated by those skilled in the

1 art that other types of computer readable media which can store data that is
2 accessible by a computer, such as magnetic cassettes, flash memory cards, digital
3 video disks, random access memories (RAMs) read only memories (ROM), and
4 the like, may also be used in the exemplary operating environment.

5 A number of program modules may be stored on the hard disk, magnetic
6 disk 160, optical disk 164, ROM 150, or RAM 152, including an operating system
7 170, one or more application programs 172, other program modules 174, and
8 program data 176. A user may enter commands and information into computer
9 142 through input devices such as keyboard 178 and pointing device 180. Other
10 input devices (not shown) may include a microphone, joystick, game pad, satellite
11 dish, scanner, or the like. These and other input devices are connected to the
12 processing unit 144 through an interface 182 that is coupled to the system bus. A
13 monitor 184 or other type of display device is also connected to the system bus
14 148 via an interface, such as a video adapter 186. In addition to the monitor,
15 personal computers typically include other peripheral output devices (not shown)
16 such as speakers and printers.

17 Computer 142 operates in a networked environment using logical
18 connections to one or more remote computers, such as a remote computer 188.
19 The remote computer 188 may be another personal computer, a server, a router, a
20 network PC, a peer device or other common network node, and typically includes
21 many or all of the elements described above relative to computer 142, although
22 only a memory storage device 190 has been illustrated in Fig. 2. The logical
23 connections depicted in Fig. 2 include a local area network (LAN) 192 and a wide
24 area network (WAN) 194. Such networking environments are commonplace in
25 offices, enterprise-wide computer networks, intranets, and the Internet. In the

1 described embodiment of the invention, remote computer 188 executes an Internet
2 Web browser program such as the "Internet Explorer" Web browser manufactured
3 and distributed by Microsoft Corporation of Redmond, Washington.

4 When used in a LAN networking environment, computer 142 is connected
5 to the local network 192 through a network interface or adapter 196. When used
6 in a WAN networking environment, computer 142 typically includes a modem 198
7 or other means for establishing communications over the wide area network 194,
8 such as the Internet. The modem 198, which may be internal or external, is
9 connected to the system bus 148 via a serial port interface 168. In a networked
10 environment, program modules depicted relative to the personal computer 142, or
11 portions thereof, may be stored in the remote memory storage device. It will be
12 appreciated that the network connections shown are exemplary and other means of
13 establishing a communications link between the computers may be used.

14 Generally, the data processors of computer 142 are programmed by means
15 of instructions stored at different times in the various computer-readable storage
16 media of the computer. Programs and operating systems are typically distributed,
17 for example, on floppy disks or CD-ROMs. From there, they are installed or
18 loaded into the secondary memory of a computer. At execution, they are loaded at
19 least partially into the computer's primary electronic memory. The invention
20 described herein includes these and other various types of computer-readable
21 storage media when such media contain instructions or programs for implementing
22 the steps described below in conjunction with a microprocessor or other data
23 processor. The invention also includes the computer itself when programmed
24 according to the methods and techniques described below. Furthermore, certain
25 sub-components of the computer may be programmed to perform the functions

1 and steps described below. The invention includes such sub-components when
2 they are programmed as described. In addition, the invention described herein
3 includes data structures, described below, as embodied on various types of
4 memory media.

5 For purposes of illustration, programs and other executable program
6 components such as the operating system are illustrated herein as discrete blocks,
7 although it is recognized that such programs and components reside at various
8 times in different storage components of the computer, and are executed by the
9 data processor(s) of the computer.

11 **Searching Data Streams**

12 As shown in Fig. 1, a network system in accordance with the invention
13 includes network server(s) 100 from which a plurality of composite media streams
14 are available. In some cases, the composite media streams are actually stored by
15 server(s) 100. In other cases, server(s) 100 obtains the composite media streams
16 from other network sources or devices.

17 The system also includes network clients 104. Generally, the network
18 clients are responsive to user input to select or request identified multimedia
19 presentations. In response to a request for a multimedia presentation, server(s)
20 100 streams the requested media stream(s) to the network client in accordance
21 with some known format such as ASF. The client renders the data streams to
22 produce the multimedia presentation.

23 Fig. 3 illustrates an exemplary client-server relationship for streaming data.
24 A network server 100 is illustrated streaming a composite media stream 202 to a
25 client 104. Alternatively, multiple servers 100 could be streaming individual or

1 composite media streams to client 104. Additional control information 203 is also
2 communicated between server 100 and client 104 to manage the streaming of
3 composite media stream 202 to client 104.

4 When a user requests a particular composite media stream, client 104
5 requests the underlying media streams from the appropriate server(s) 100. This
6 request can be from a standalone control application that is stored and executed at
7 client 104, or alternatively an application that is hosted at a server 100 and is
8 transmitted to client 104 for execution. For example, the control application could
9 be hosted in a HTTP web page (maintained by either server 100 or another server
10 coupled to the network) in accordance with Hypertext Markup Language (HTML)
11 or Extended Markup Language (XML). The control application (whether
12 standalone or server-hosted) includes identifier(s) of the composite media stream
13 and/or the individual media streams of the multimedia presentation, and
14 coordinates when and how they are presented at client 104.

15 Each media stream has a timeline, and the timelines of the individual
16 streams are synchronized with each other so that the streams can be rendered in
17 combination to produce coordinated multimedia content at the network client 104.
18 A streaming module 205 in server 100 manages the streaming of the composite
19 media stream to client 104 based at least in part on the delivery times of the data
20 units in the composite media stream.

21 The client computer has a demultiplexer component 204 that receives the
22 composite media stream and separates out the individual media streams from the
23 composite format in which the data is streamed (such as ASF). This results in a
24 video stream 206, an audio stream 208, a text stream 210, an image stream 212,
25 and an animation stream 214.

1 The individual media streams are received from demultiplexer 204 by
2 respective decoders 222, 224, 226, 228, and 230 that perform in accordance with
3 the particular data format being employed. For example, the decoders might
4 perform data decompression. The decoded streams are then provided to and
5 received by respective renderers 234, 236, 238, 240, and 242. The rendering
6 components 234 – 242 render the streams as the streams continue to be streamed
7 from the network server 100.

8 Server 100 stores a composite media stream 248 (e.g., in accordance with
9 ASF) including multiple individual media streams 250 – 258 for a multimedia
10 presentation. The individual media streams are of different types, which in the
11 illustrated embodiment are audio stream 250, video stream 252, image stream 254,
12 text stream 256, and animation stream 258. A streaming module 205 manages, on
13 behalf of server 100, the communication between server 100 and client 104,
14 including the streaming of the composite media stream 202 and communication of
15 control information to and from client 104.

16 Server 100 also stores indexes 260 – 268, each corresponding to one of the
17 individual media streams 250 – 258. Indexes 260 – 268 can be part of the same
18 composite media stream 248 as the individual media streams 250 – 258, or
19 alternatively may be stored separately. In the illustrated example, each individual
20 media stream 250 – 258 has a corresponding index 260 – 268, illustrated as audio
21 index 260, video index 262, image index 264, text index 266, and animation index
22 268. Alternatively, one or more of the indexes 260 – 268 may be combined
23 together.

24 Alternatively, some or all of indexes 260 – 268 may be stored at a remote
25 server, such as index server 102 of Fig. 1. Or, in other alternate embodiments,

1 some or all of indexes 260 – 268 may be stored at client 104 or transferred to
2 client 104 for searching.

3 Each of the indexes 260 – 268 maintains a correspondence between a
4 particular term or element of the associated media stream and a temporal location
5 of the associated media stream. This correspondence identifies, for each term or
6 element, a temporal location(s) of the associated media stream at which that term
7 or element occurs. This correspondence can be maintained by storing multiple
8 entries in the index, each entry including a term or element of the associated media
9 stream and a temporal location(s) at which that term or element occurs. In one
10 implementation these terms or elements are characters, words, symbols, or groups
11 thereof. Alternatively the exact nature of these terms or elements may be
12 dependent on the nature of the associated media stream. For example, text index
13 266 may include words or phrases as terms or elements, while audio index 260
14 may include digital representations of audio waveforms as terms or elements.

15 Indexes 260 – 268 can be generated in any of a wide variety of manners,
16 including both manual and automatic generation. Manual generation can be
17 performed by an individual (e.g., the author of the multimedia presentation)
18 manually identifying the different terms or elements to index for each of the
19 individual media streams and the temporal location(s) of each of these terms or
20 elements. Automatic generation of the indexes can be performed by server 100 or
21 another device. The manner in which the automatic generation is carried out is
22 dependent on the nature of the associated individual media stream.

23 Text streams can be indexed based on different elements, such as
24 characters, symbols, words, or groups thereof (e.g., phrases or sentences). In the
25 illustrated example, server 100 (or other device generating the index) generates the

1 index by identifying each of the elements in the text stream and their
2 corresponding presentation times. As each element can occur multiple times in a
3 text stream, multiple presentation times may be identified for the index.

4 Image streams can be indexed in a similar manner as text streams. Any of a
5 variety of conventional pattern recognition techniques can be used to identify
6 particular objects in the image stream or different characteristics of those objects
7 (e.g., color). A textual description of each of these objects is included as an
8 element in the index, along with its corresponding presentation time.

9 Animation streams can be indexed in an analogous manner as image
10 streams. However, each object in the animation may have a range of presentation
11 times corresponding to, for example, the object moving or changing locations over
12 time. Thus, multiple presentation times (the presentation times for this range) may
13 be associated with the object. The earliest presentation time in the range may be
14 used in the index as the temporal location for this object, or alternatively another
15 temporal location within the range may be used (e.g., the entire range, or
16 alternatively a mid-range or "average" temporal range).

17 Audio streams can be similarly indexed. Conventional audio analysis
18 techniques may be used to identify words or groups of words in the audio stream,
19 based on the digital representation of the analog waveform of the audio content.
20 These digital representations of words or groups of words are stored in the index
21 along with their temporal locations in the audio content. Alternatively,
22 conventional speech to text techniques can be used to convert the audio stream to
23 words which can then be included in the index analogous to text streams discussed
24 above.

1 Video streams can also be indexed. Using image analysis techniques
2 similar to those for analyzing image streams, different objects within the video
3 stream can be identified and included in the index. Alternatively, general video
4 frame characteristics can be indexed, such as predominant colors within the frame.
5 Objects in a video stream may correspond to a range of presentation times
6 analogous to those in animation streams, and the temporal locations of these can
7 be stored in the index in a manner analogous to those of animation streams.

8 A user at client 104 can search through the media streams 250 – 258 by
9 submitting a search request and search criteria (e.g., particular words, or other text
10 or symbols) to server 102. A search engine 270 receives such search requests and
11 accompanying search criteria at server 102. Search engine 270, upon receipt of
12 the search request, compares the search criteria to the entries in each of indexes
13 260 – 268 to determine whether any of the entries match or are satisfied by the
14 search criteria. Thus, a single search request from the user can initiate searching
15 of multiple individual media streams. Alternatively, the user may identify in the
16 search request specific media streams that are to be searched, with the other media
17 streams to be left unsearched.

18 Additionally, search engine 270 can search multiple different individual or
19 composite streams located at server 100, regardless of whether the streams
20 correspond to the same or different media presentations. Alternatively, search
21 engine 270 may forward a search request to other servers 100 to identify matches
22 with media streams stored at such servers. Thus, a single search request from the
23 user can initiate searching of media streams of multiple different media
24 presentations regardless of where the streams are stored.

1 Alternatively, search engine 270 may not use indexes 260 – 268. Rather,
2 upon receipt of a search request search engine 270 can access the various streams
3 250 – 258 directly to identify terms or elements to compare to the search criteria.
4 These terms or elements can be identified in any of the manners discussed above
5 with reference to generating the indexes.

6 Situations can arise where the search criteria match multiple entries in the
7 index. For example, a particular object may occur multiple times in an image
8 stream and thus have multiple temporal locations associated with it in the index.
9 Search engine 270 can identify one of these temporal locations to use as the result
10 of the search process. In one implementation, if the media stream is currently
11 being played back, then the current presentation time of the multimedia
12 presentation is identified (e.g., from client 104). Search engine 270 then selects
13 the next temporal location associated with the index entry that is after the current
14 presentation time. Alternatively, search engine 270 may select the temporal
15 location that is closest to the current presentation time as the result, or alternatively
16 may use some other process for identifying one of the presentation times.
17 Alternatively, search engine 270 may make such determinations based on the
18 delivery times for the data units of the media stream rather than the presentation
19 times.

20 Search engine 270 can take a variety of different actions when a term or
21 element in the index matches the search criteria. For example, the search engine
22 270 may inform streaming module 205 to “seek” to that temporal location and
23 begin streaming the media stream to client 104 at that temporal location. By way
24 of another example, the matching entry and associated temporal locations) may be
25 returned to client 104 and displayed to the user.

1 Additionally, the search process may make use of global variables. For
2 example, a global character can be used to represent one or more characters,
3 symbols, or words during the searching process.

4 Fig. 4 is a flowchart illustrating an exemplary process for searching media
5 streams in accordance with one implementation of the invention. The process of
6 Fig. 4 may be performed in software, firmware, hardware, or any combination
7 thereof. Fig. 4 is described with additional reference to components in Figs. 1 and
8 3.

9 An index for one of the media streams of a media presentation is initially
10 generated, step 302. Indexes for any additional media streams are also generated,
11 steps 304 and 302. This index generation can be carried out by multimedia server
12 100, index server 102, or alternatively some other device (not shown) and
13 subsequently transferred to server 100 or 102.

14 Once the indexes are generated, server 100 eventually receives a search
15 request with accompanying search criteria, step 306. Search engine 270 compares
16 the index(es) of the media stream(s) corresponding to the search request to the
17 search criteria, step 308, and attempts to identify a match, step 310. If no match is
18 identified, then an indication that the search failed is sent to the client, step 312.
19 Alternatively, steps 308 – 310 may be carried out at an index server 102 or client
20 104 rather than media server 102.

21 However, if a match is identified then search engine 270 identifies a
22 temporal location corresponding to the match (selecting one of multiple temporal
23 locations if necessary), step 314. Streaming module 205 then streams, to client
24 104, the data for the media presentation starting at a location based on the
25 identified temporal location, step 316. Streaming module 205 may stream the data

1 beginning at the identified temporal location, or alternatively may “rewind” or
2 “back up” to a temporal point prior to the temporal location. This “rewinding”
3 may be of a fixed amount (e.g., three seconds), or alternatively may be based on
4 pauses or breaks. For example, conventional audio or video analysis programs
5 may be used to identify pauses or breaks in the speech or action in the multimedia
6 presentation, and module 205 may search back through the multimedia
7 presentation beginning at the identified temporal location to identify such a pause
8 or break and begin streaming at that location. Alternatively, rather than “seeking”
9 to the match location in step 316, other indications of a match may be provided to
10 client 104.

11 In alternative implementations, the index generation of steps 302 – 304 is
12 performed in response to a user search request rather than as an initialization
13 process as illustrated in Fig. 4. For example, when a user submits a request for a
14 multimedia presentation that has not been indexed, the index generation may occur
15 in response to the request (either generating all of the indexes or only those
16 necessary until a match with the search criteria is reached).

17 Alternatively, rather than performing any sort of “rewind” or “backing up”
18 process during the search request, such rewinding can be performed at the time the
19 index is generated. Thus, rather than storing a specific presentation time that
20 corresponds to the time that a particular term or object occurs in the multimedia
21 presentation, the index could store the presentation times that should be used to
22 begin playback of the multimedia presentation in the event of a match to the
23 particular term or object.

24 Fig. 5 is a flowchart illustrating another exemplary process for searching
25 media streams in accordance with another implementation of the invention which

1 does not use indexes. The process of Fig. 5 may be performed in software,
2 firmware, hardware, or any combination thereof. Fig. 5 is described with
3 additional reference to components in Figs. 1, 3 and 4.

4 A search request and corresponding search criteria are initially received by
5 server 100, step 322. Search engine 270 then compares the data of the media
6 streams corresponding to the search request to the search criteria, step 324. If no
7 match is identified, step 328, then an indication that the search has failed is sent to
8 client 104, step 328. However, if a match is identified, then search engine 270
9 identifies the temporal location in the media streams corresponding to the match,
10 step 330. Streaming module 205 then begins streaming the media presentation to
11 the client starting at the match location, step 332, analogous to step 316 of Fig. 4.

12 The comparisons performed by search engine 270 in steps 308 of Fig. 4 and
13 324 of Fig. 5 can be carried out in a variety of different manners. In one
14 implementation, the comparisons are made through each of the indexes (or media
15 streams) to identify all possible matches to the search criteria and then the
16 temporal location of one of these matches is selected in step 314 (Fig. 4) or 330
17 (Fig. 5). Alternatively, all of these possible matches may be provided to the user
18 (e.g., the presentation times) and the user can select one of them to “seek” to.
19 Alternatively, as soon as one match of the search criteria is found the comparison
20 can stop and that temporal location sought to. Alternatively, search engine 270
21 may identify the current presentation time of the multimedia presentation and
22 search, from that temporal location on, for the next presentation time in each of the
23 media streams that satisfies the search request and select from this set of
24 presentation times.
25

Recording Data Streams

Returning to Fig. 3, client 104 also includes a stream saving module 272. Module 272 stores composite media stream 202 locally at client 104 as stream 202 is received. Module 272 also optionally receives indexes 260 – 268 and stores them locally at client 104 as well. This storage can be done concurrently with the rendering of the media streams, or alternatively can be carried out independently without rendering the streams. Module 272 can store the media streams locally in response to a user request to store the streams, or alternatively automatically in response to some other event or action (e.g., an indication of search criteria being satisfied from server 100).

Module 272 also generates and stores a markup document that describes how the various media streams are to be rendered (e.g., the screen locations for audio, text, image, and animation streams). This markup document can be generated using any of a variety of conventional programming languages, such as HTML or XML. In the illustrated example, module 272 generates the markup document by modifying a pre-existing markup document, such as one received from server 100 for the rendering of the individual media streams 250 – 258. Module 272 modifies the pre-existing markup document by searching the pre-existing document for references to the locations of media streams 250 – 258 and changing those references to the locally stored media streams. Alternatively, module 272 may generate such a markup document “from scratch”.

Module 272 may also “package” the locally stored media streams and the modified markup document into the same file or folder, thereby allowing easier transport of the files. It is to be appreciated that the locally stored media streams could be transferred or copied to another client and played back without requiring

1 further modification to the markup document so long as the path names where the
2 files are stored on the different clients remain the same.

3 Figs. 6 and 7 are block diagrams illustrating the local storage of a
4 multimedia presentation in accordance with one implementation of the invention.
5 In Figs. 6 and 7, a multimedia presentation is shown including two video streams,
6 an image stream, an audio stream, a text stream, and an animation stream.
7 Initially, these streams are stored at two different remote multimedia servers 340
8 and 342. A markup document 344 of Fig. 6 at client 104 includes references to or
9 identifiers of the various individual media streams, illustrated as identifiers 346 –
10 356. As shown, each of the identifiers 346 – 356 identifies one of the remotely
11 stored media streams 358 – 368, respectively.

12 To store the multimedia presentation locally, the media streams 358 – 368
13 are streamed (or otherwise copied) to local storage 374 of Fig. 7. Markup
14 document 344 is also modified to generate markup document 376 that references
15 the locally stored media streams 378 – 388. By changing the references to the
16 locally stored media streams 378 – 388, subsequent playback of the multimedia
17 presentation using markup document 376 will result in the locally stored streams
18 378 – 388 being played back (being input to either demultiplexer 204 or the
19 decoders 222 – 230 directly) rather than the remotely stored streams 358 – 368 of
20 Fig. 6, thereby avoiding any access to remote servers.

21 Fig. 8 is a flowchart illustrating an exemplary process for recording a
22 multimedia presentation in accordance with one implementation of the invention.
23 The process of Fig. 8 is performed by client 104 of Fig. 3 and may be performed
24 in software, firmware, hardware, or any combination thereof. Fig. 8 is described
25 with additional reference to components in Fig. 3.

1 Client 104 initially receives a markup document referencing one or more
2 media streams of a media presentation, step 390. Client 104 also receives the
3 media stream(s) of the media presentation, step 392. The received media
4 stream(s) are stored locally at client 104, step 394. Client 104 also modifies the
5 markup document received in step 390 to reference the locally stored media
6 stream(s), step 396, and stores the modified markup document locally, step 398.
7 Client 104 may also optionally package the locally stored media stream(s) and
8 modified markup document into a single file or folder, allowing for easy
9 subsequent transfer of the files.

10 11 **Conclusion**

12 The invention allows for searching and recording of streaming multimedia
13 data. Any or all of the multiple data streams of a multimedia presentation can
14 advantageously be searched via a single search request by a user, and the user can
15 be immediately presented with the temporal location of the multimedia
16 presentation that satisfies his or her search request. Additionally, a streaming
17 multimedia presentation can advantageously be saved locally, allowing subsequent
18 playback of the presentation when not connected to the remote multimedia servers.

19 Although the invention has been described in language specific to structural
20 features and/or methodological steps, it is to be understood that the invention
21 defined in the appended claims is not necessarily limited to the specific features or
22 steps described. Rather, the specific features and steps are disclosed as preferred
23 forms of implementing the claimed invention.

CLAIMS

1. A method of searching streaming media presentations, the method comprising:

receiving a search request including search criteria;
determining a temporal location in a streaming media presentation that corresponds to data of the media presentation that matches the search criteria; and
returning an indication of the temporal location to a source of the request.

2. A method as recited in claim 1, further comprising saving a media data stream of the media presentation locally at a client computer if data in the media presentation matches the search criteria, otherwise not saving the media data stream locally at the client computer.

3. A method as recited in claim 1, wherein the determining comprises checking, for each of a plurality of individual media streams of the media presentation, whether data of the media stream matches the search criteria.

4. A method as recited in claim 1, wherein the determining comprises comparing data of media streams corresponding to a plurality of different media presentations to the search criteria.

5. A method as recited in claim 1, wherein the temporal location comprises a presentation time of the media presentation.

1 6. A method as recited in claim 1, wherein the media presentation
2 comprises a composite media stream including a plurality of individual media
3 streams.

4
5 7. A method as recited in claim 1, further comprising:
6 seeking to the temporal location; and
7 streaming of the media presentation to a client based on the temporal
8 location.

9
10 8. A method as recited in claim 7, wherein the streaming comprises
11 streaming the media presentation to the client beginning at the temporal location.

12
13 9. A method as recited in claim 1, wherein the returning comprises
14 displaying the indication to a user.

15
16 10. A method as recited in claim 1, wherein the receiving comprises
17 receiving the request from a client computer via a network.

18
19 11. A method as recited in claim 1, wherein the receiving comprises
20 receiving the request, at an index server, from a media server via a network.

21
22 12. A method as recited in claim 1, wherein the determining comprises:
23 accessing an index corresponding to an individual media data stream of the
24 media presentation;
25 checking whether the search criteria matches data in the index; and

1 if the search criteria matches data in the index, then identifying a
2 presentation time of the media presentation at which the search criteria are
3 satisfied.

4
5 **13.** A method as recited in claim 1, wherein the search criteria
6 comprises user-specified criteria.

7
8 **14.** A method as recited in claim 1, wherein the returning comprises
9 sending the indication from an index server to a media server that is a source of at
10 least part of the media presentation.

11
12 **15.** One or more computer-readable memories containing a computer
13 program that is executable by a processor to perform the method recited in claim
14 1.

15
16 **16.** An apparatus comprising:
17 a memory device to store a plurality of multimedia data streams
18 corresponding to a streaming multimedia presentation; and
19 a search engine, coupled to the memory device, to,
20 receive a search request corresponding to the multimedia
21 presentation,
22 determine whether any of the multimedia data streams
23 corresponding to the multimedia presentation satisfy search criteria
24 corresponding to the search request, and
25

1 return an indication of whether any of the multimedia data streams
2 satisfy the search criteria.

3
4 **17.** An apparatus as recited in claim 16, wherein the memory device
5 comprises a random access memory.

6
7 **18.** An apparatus as recited in claim 16, wherein the apparatus
8 comprises a multimedia server and the search engine is to receive the search
9 request from a client computer via a network.

10
11 **19.** An apparatus as recited in claim 16, wherein the apparatus
12 comprises a multimedia server and the search engine is to determine whether any
13 of the multimedia data streams satisfy the search criteria by forwarding the search
14 criteria to an index server.

15
16 **20.** An apparatus as recited in claim 16, wherein the apparatus
17 comprises a client computer and the search engine is to receive a search request
18 from a user of the client computer.

19
20 **21.** An apparatus as recited in claim 16, wherein the apparatus is to
21 determine whether any of the multimedia data streams satisfy the search criteria by
22 comparing, for each of the multimedia data streams, the search criteria to index
23 data for the multimedia data stream.

1 **22.** An apparatus as recited in claim 16, wherein the apparatus is to
2 determine whether any of the multimedia data streams satisfy the search criteria by
3 comparing, for each of the multimedia data streams, the search criteria to the data
4 of the multimedia data stream.

5
6 **23.** An apparatus as recited in claim 16, wherein:
7 the apparatus further comprises a streaming component to manage
8 streaming of the multimedia data streams to a client computer;
9 the search engine is to identify a temporal location of the multimedia data
10 streams that satisfies the search criteria and forward the temporal location to the
11 streaming component; and
12 the streaming component is to stream the multimedia data streams to the
13 client computer at a beginning temporal location based on the identified temporal
14 location.

15
16 **24.** An apparatus as recited in claim 16, wherein the apparatus further
17 comprises a data saving component to receive the multimedia data streams from a
18 multimedia server and store the multimedia data streams in the memory device.

19
20 **25.** A system comprising:
21 a client computer, coupled to a network, to receive streaming data via the
22 network; and
23 a multimedia server, coupled to the network, to stream the streaming data to
24 the client computer, the multimedia server including one or more index files
25

1 corresponding to the streaming data and a search engine to check whether data in
2 the index files matches search criteria received from the client computer.

3
4 **26.** A system as recited in claim 25, wherein the client computer
5 comprises a demultiplexer to separate the streaming data into individual media
6 streams, and a data saver to save the individual media streams at the client
7 computer.

8
9 **27.** A system comprising:
10 a client computer, coupled to a network, to receive streaming data via the
11 network;

12 a multimedia server, coupled to the network, to stream the streaming data to
13 the client computer; and

14 an index server, coupled to the network, to store index files corresponding
15 to the streaming data and to check, upon receipt of a search request, whether any
16 portion of the streaming data matches search criteria of the search request based at
17 least in part on the contents of the index files.

18
19 **28.** A method comprising:
20 identifying a set of search criteria to be compared to data of a streaming
21 media presentation;

22 transmitting the set of search criteria to a server; and

23 receiving an indication of whether the search criteria match any portion of
24 the streaming media presentation.

1 **29.** A method as recited in claim 28, wherein the receiving an indication
2 comprises receiving the streaming media presentation beginning at a temporal
3 location corresponding to a portion of the streaming media presentation that
4 matches the search criteria.

5
6 **30.** A method as recited in claim 28, wherein the transmitting comprises
7 transmitting the set of search criteria to an index server.

8
9 **31.** A method as recited in claim 28, further comprising storing the
10 streaming media presentation locally.

11
12 **32.** One or more computer-readable memories containing a computer
13 program that is executable by a processor to perform the method recited in claim
14 28.

15
16 **33.** A method comprising:
17 receiving a plurality of media streams as streaming data from a multimedia
18 server;
19 storing the plurality of media streams locally; and
20 generating a markup document describing how the plurality of media
21 streams are to be presented and referencing the locally stored plurality of media
22 streams.

1 **34.** A method as recited in claim 33, wherein the receiving the plurality
2 of media streams comprises receiving the plurality of streams as a composite
3 media stream.
4

5 **35.** A method as recited in claim 33, wherein the generating comprises:
6 receiving, from the multimedia server, an original markup document
7 referencing the plurality of media streams stored at the multimedia server; and
8 modifying the original markup document to reference the plurality of
9 locally stored media streams rather than the plurality of media streams stored at
10 the multimedia server.
11

12 **36.** A method as recited in claim 33, further comprising:
13 receiving a search request with search criteria; and
14 accessing the locally stored plurality of media streams to determine whether
15 the search criteria is satisfied by a portion of the plurality of media streams.
16

17 **37.** A method as recited in claim 33, further comprising:
18 receiving a plurality of index files corresponding to the plurality of media
19 streams; and
20 storing the plurality of index files locally.
21

22 **38.** One or more computer-readable memories containing a computer
23 program that is executable by a processor to perform the method recited in claim
24 33.
25

1 **39.** One or more computer-readable media having stored thereon a
2 computer program that, when executed by one or more processors, causes the one
3 or more processors to perform functions including:

4 receiving a markup document, from a multimedia server, that references a
5 plurality of multimedia data streams at one or more remote media servers, and that
6 identifies how the plurality of multimedia data streams are to be presented at a
7 client computer;

8 receiving the plurality of multimedia data streams from the one or more
9 remote media servers;

10 storing the plurality of multimedia data streams locally at the client
11 computer;

12 modifying the markup document to reference the plurality of locally stored
13 multimedia data streams rather than the plurality of remotely stored multimedia
14 data streams; and

15 storing the modified markup document.
16

17 **40.** One or more computer-readable media as recited in claim 39,
18 wherein the computer program further causes the one or more processors to
19 perform functions including subsequently using the modified markup document to
20 present the plurality of locally stored multimedia data streams at the client
21 computer.
22
23
24
25

1 **41.** One or more computer-readable media as recited in claim 39,
2 wherein the computer program further causes the one or more processors to
3 perform functions including combining the plurality of locally stored multimedia
4 data streams and the modified markup document into a common location.

5
6 **42.** One or more computer-readable media as recited in claim 39,
7 wherein the computer program further causes the one or more processors to
8 perform functions including:

9 receiving a plurality of index files corresponding to the plurality of
10 multimedia data streams; and
11 storing the plurality of index files locally at the client computer.

12
13 **43.** One or more computer-readable media as recited in claim 42,
14 wherein the computer program further causes the one or more processors to
15 perform functions including:

16 receiving a search request from a user;
17 checking the plurality of locally stored index files to determine whether any
18 portion of the locally stored multimedia data streams correspond to the search
19 request; and
20 indicating to the user whether any portion of the locally stored multimedia
21 data streams correspond to the search request.

1 **ABSTRACT**

2

3 In a networked client/server system, media data is streamed from a server to

4 a client. A user of the client can search the media data to identify portions that

5 satisfy certain search criteria, and/or store the media data locally at the client for

6 subsequent playback.

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

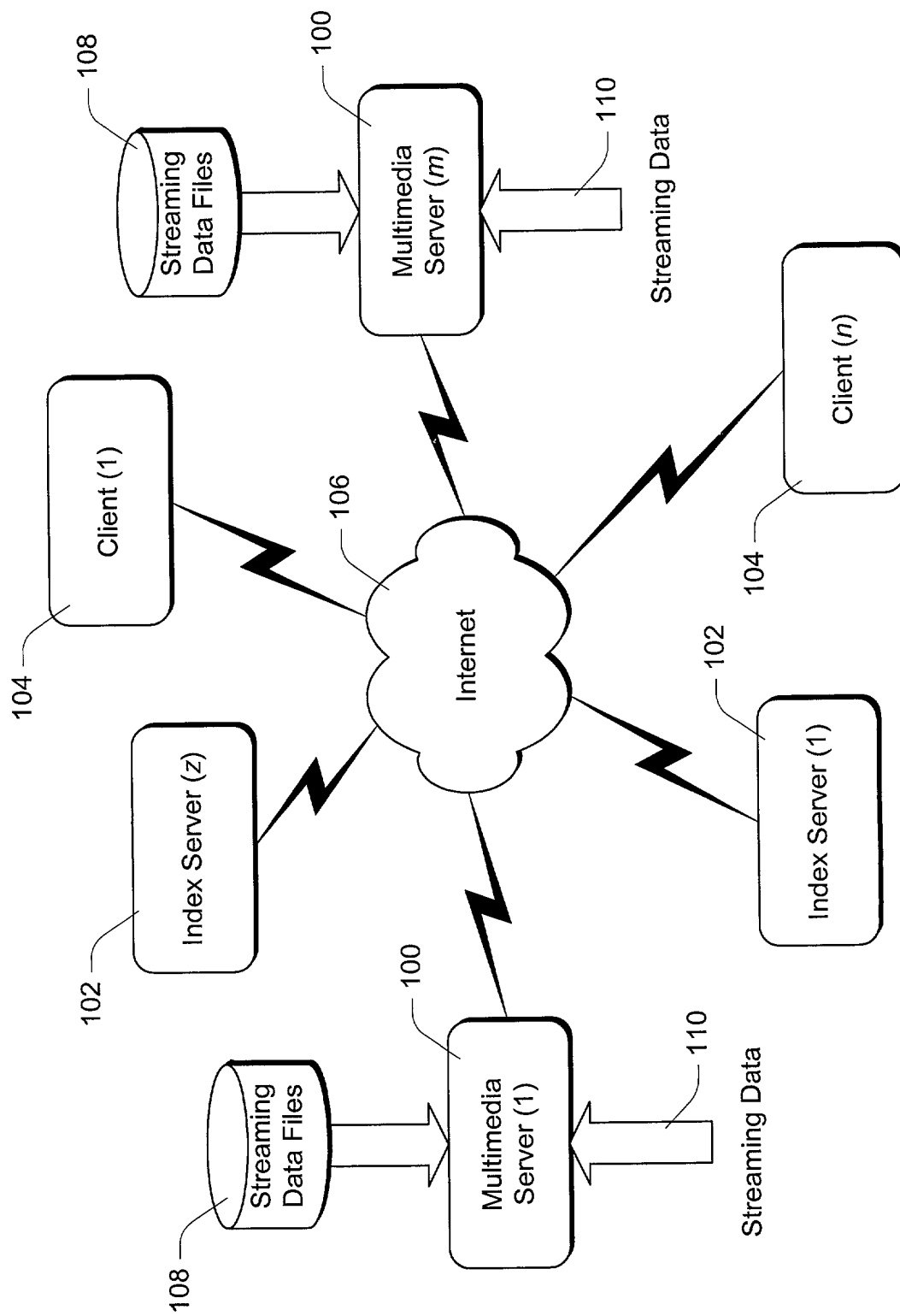
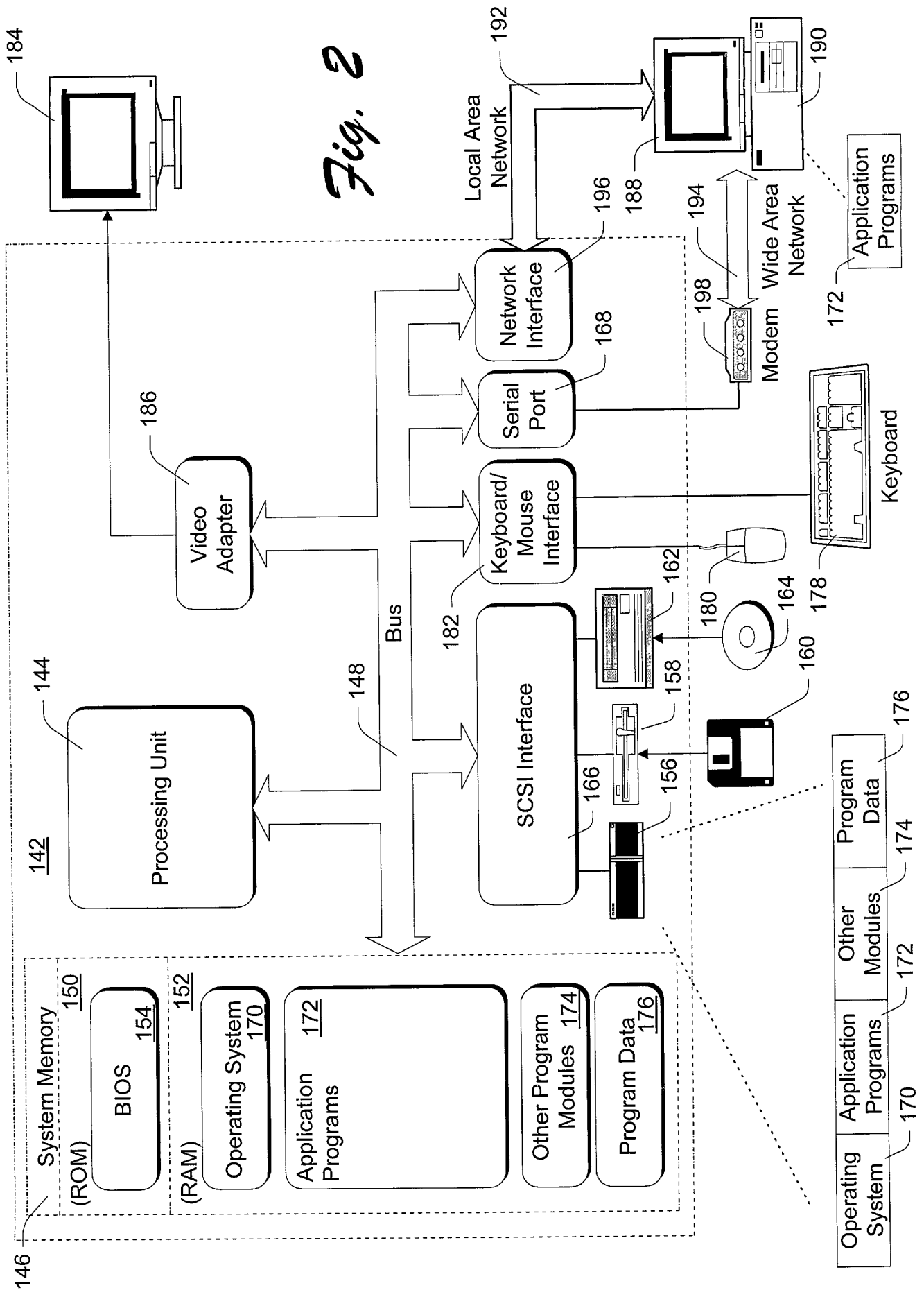
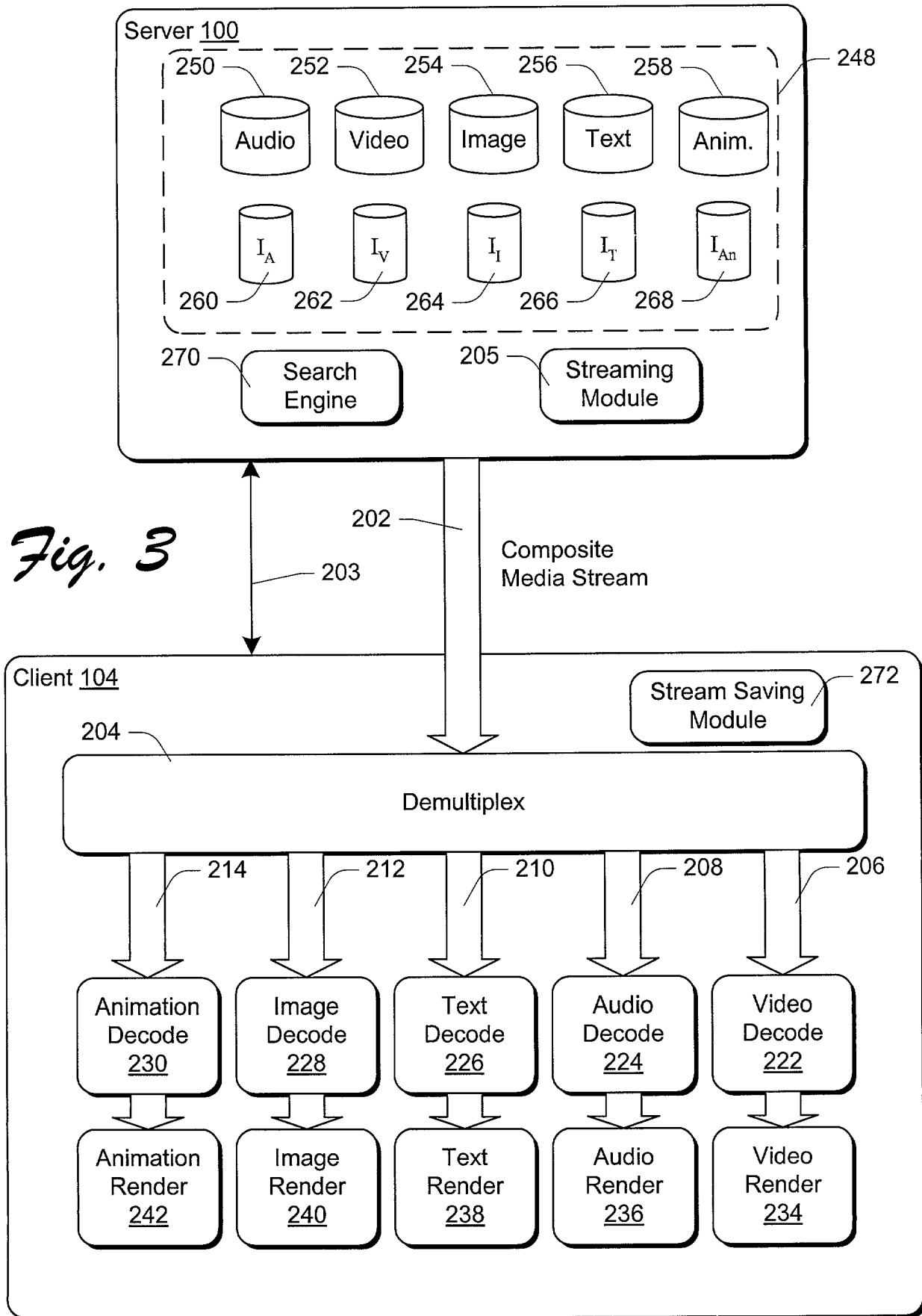
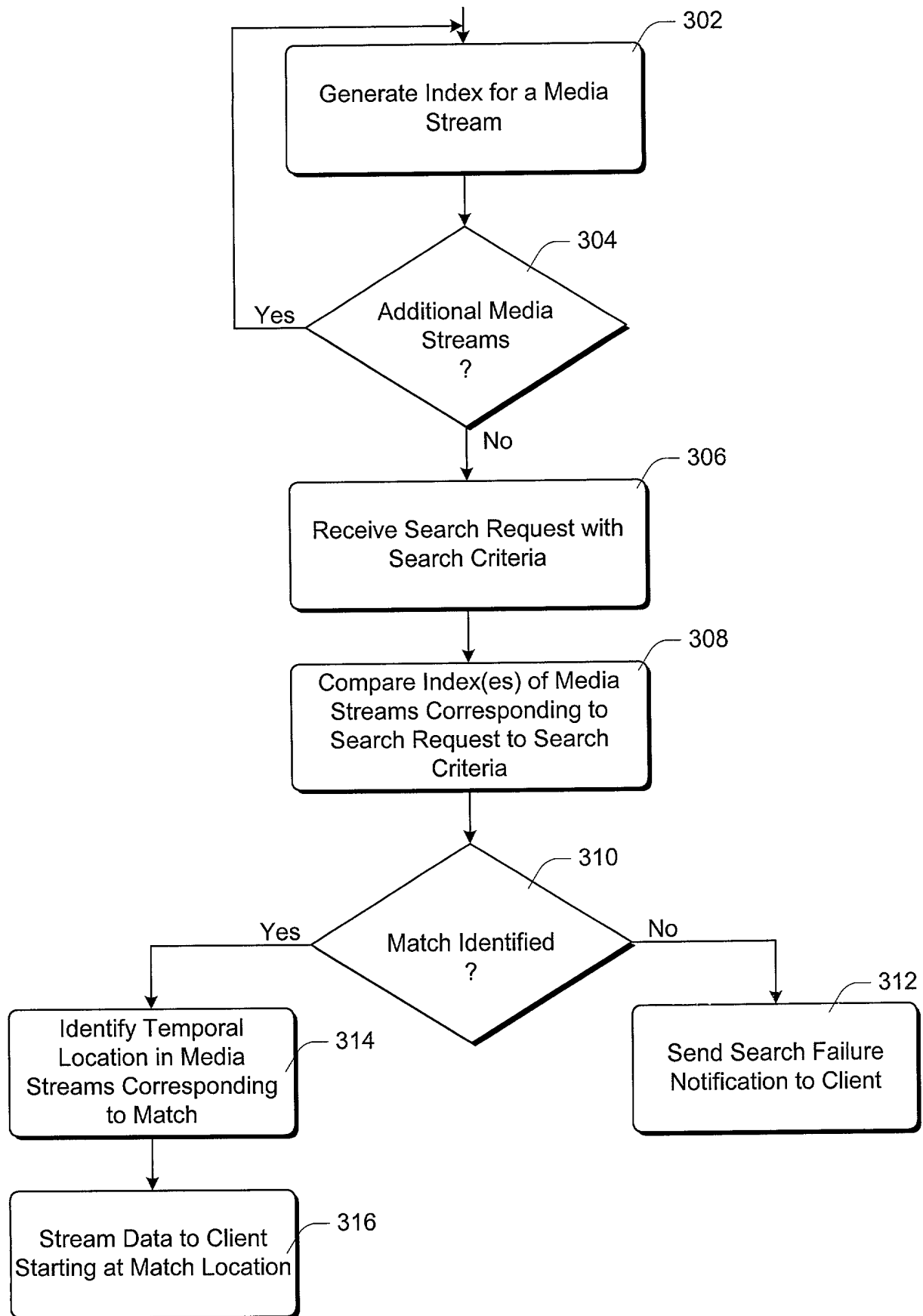
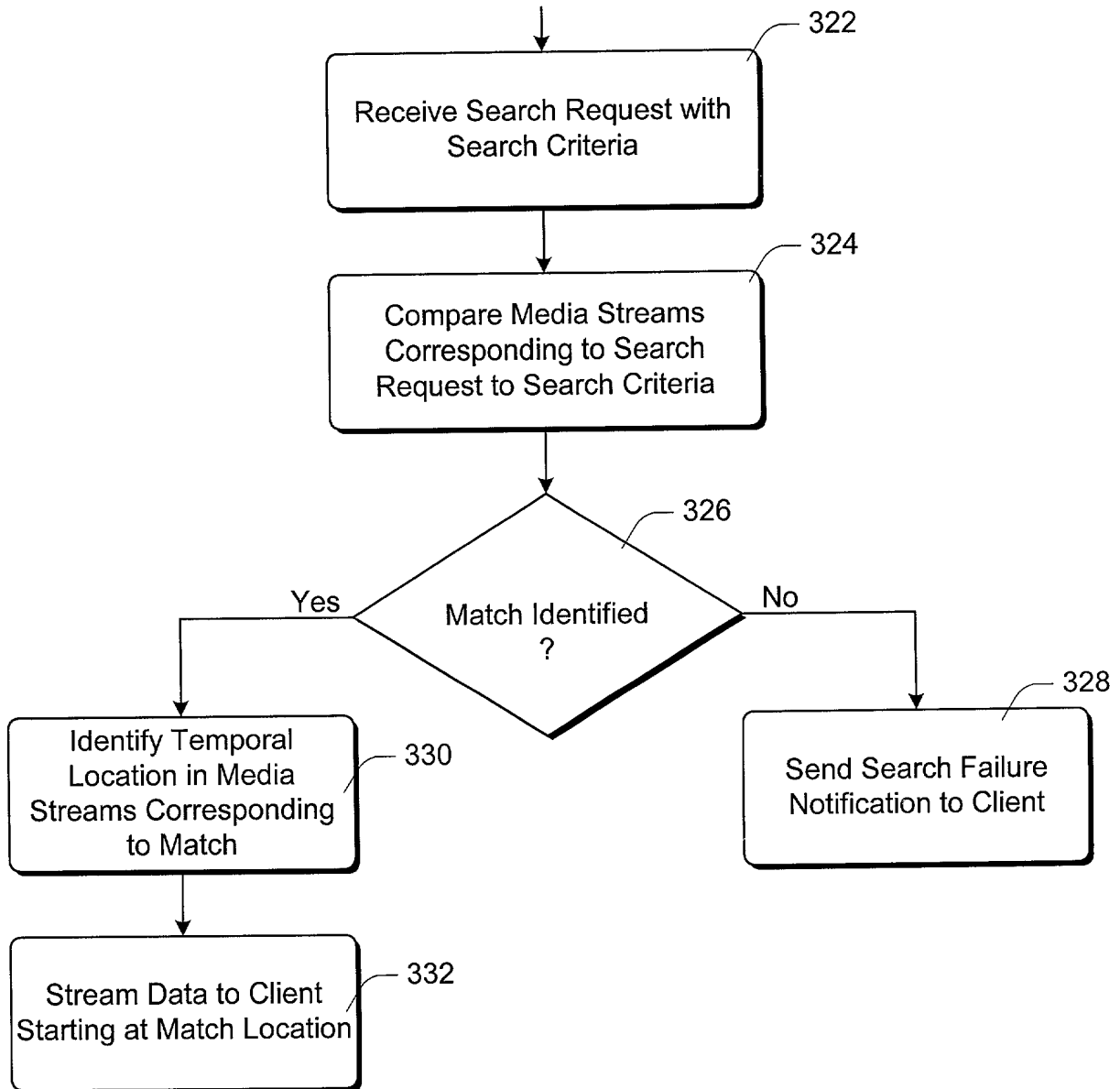
*Fig. 1*

Fig. 2





*Fig. 4*

*Fig. 5*

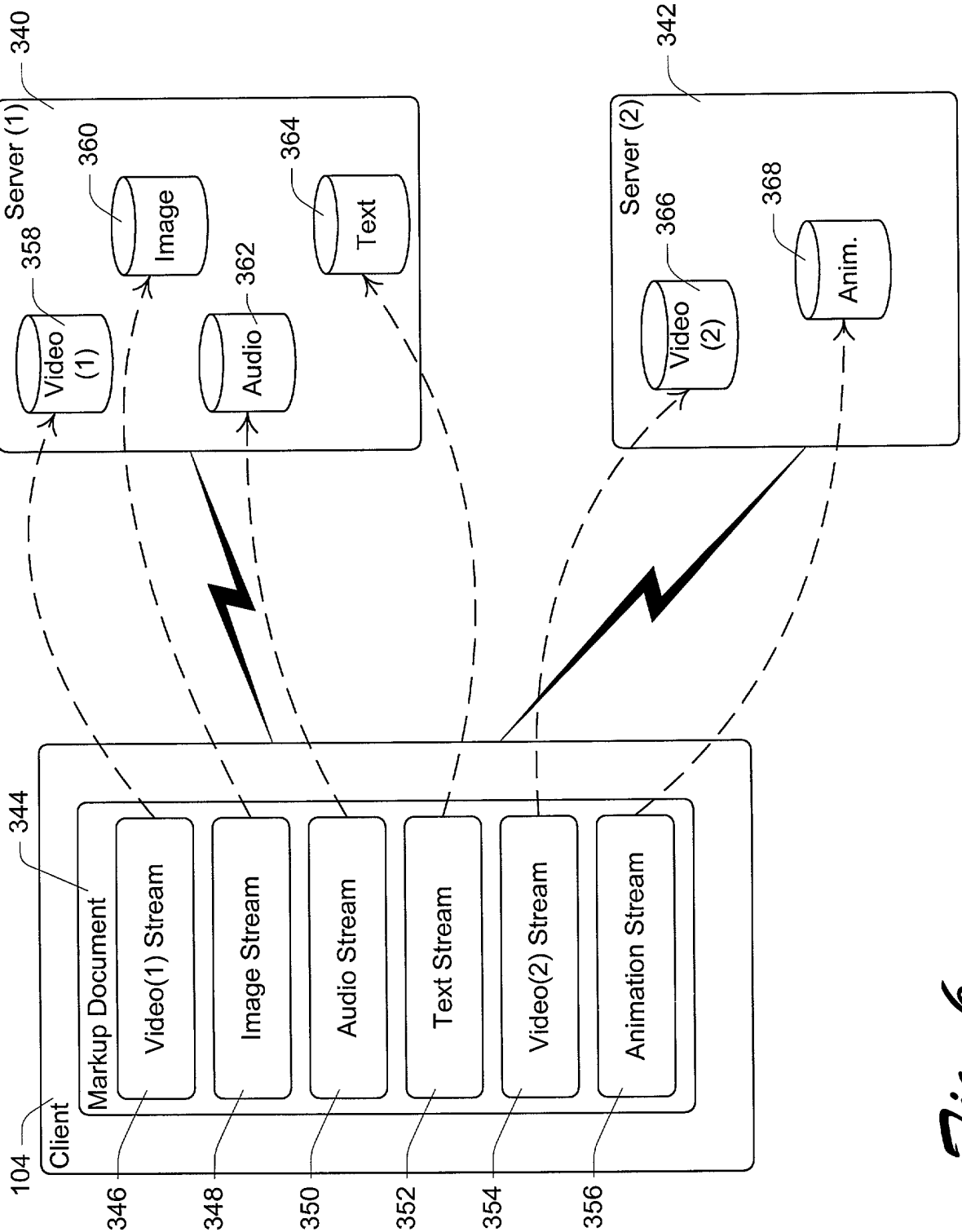


Fig. 6

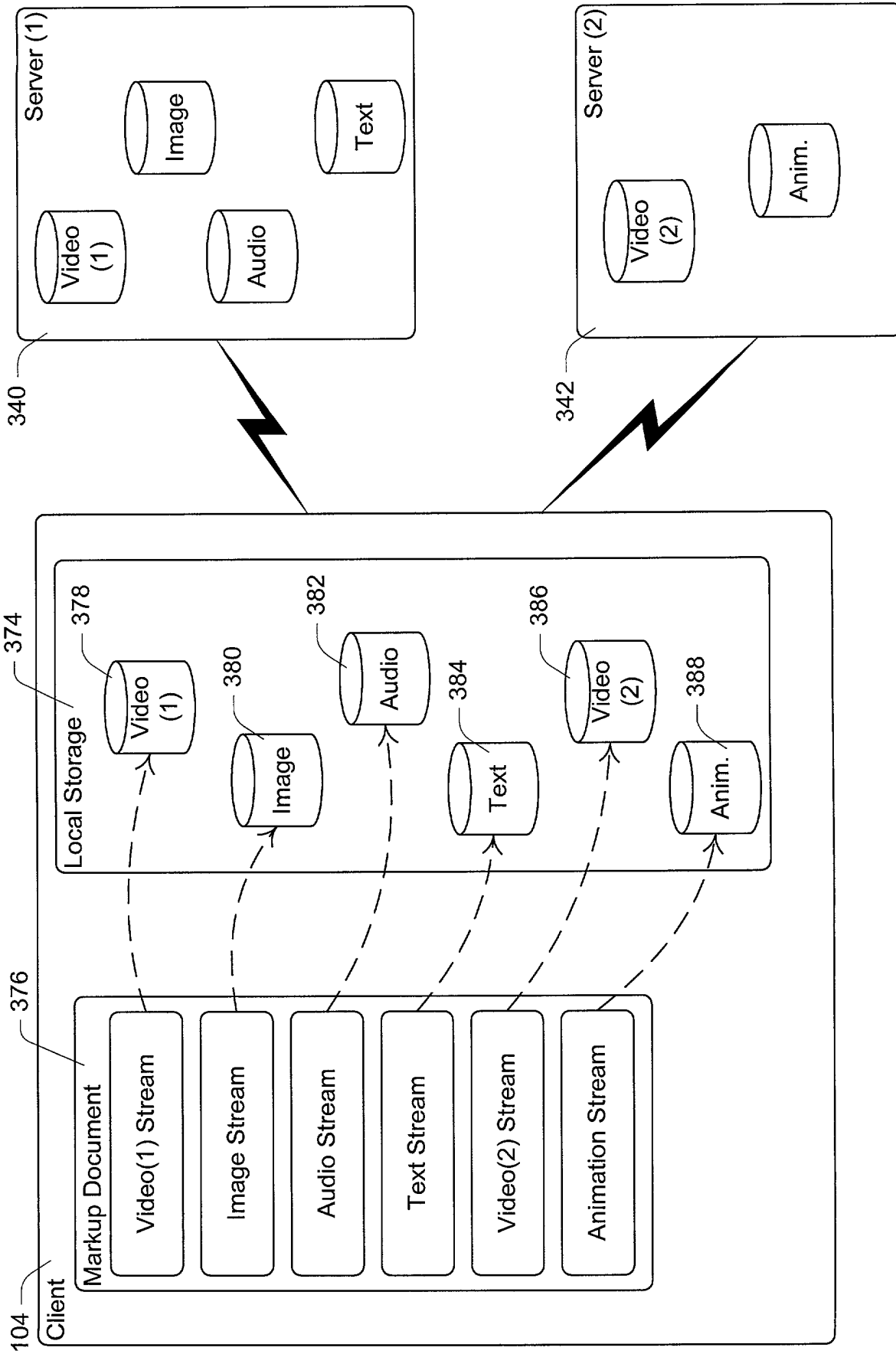
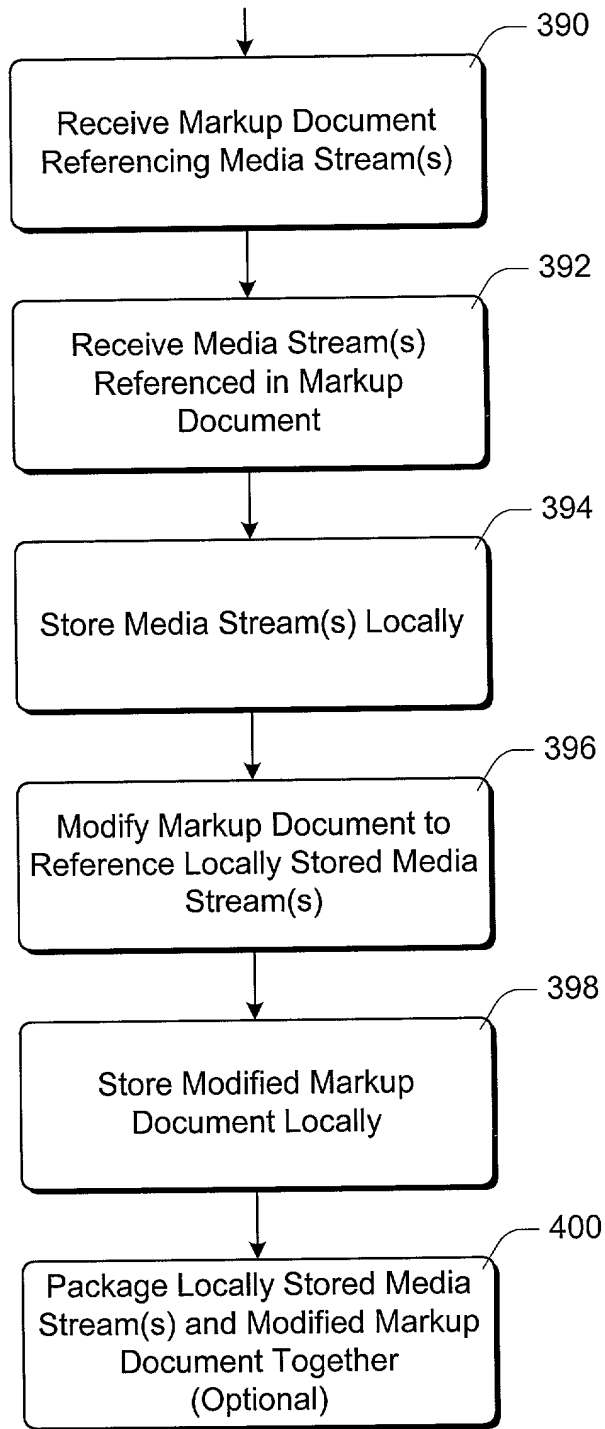


Fig. 7

*Fig. 8*

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

InventorshipOmoigui
 Applicant Microsoft Corporation
 Attorney's Docket No. MS1-362US
 Title: Searching and Recording Media Streams

DECLARATION FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name.

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled "Searching and Recording Media Streams," the specification of which is attached hereto.

I have reviewed and understand the content of the above-identified specification, including the claims.

I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, § 1.56(a).

PRIOR FOREIGN APPLICATIONS: no applications for foreign patents or inventor's certificates have been filed prior to the date of execution of this declaration.

Power of Attorney

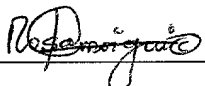
I appoint the following attorneys to prosecute this application and transact all future business in the Patent and Trademark Office connected with this application:
 Lewis C. Lee, Reg. No. 34,656; Daniel L. Hayes, Reg. No. 34,618; Allan T.

Sponseller, Reg. 38,318; Steven R. Sponseller, Reg. No. 39,384; James R. Banowsky, Reg. No. 37,773; Lance R. Sadler, Reg. No. 38,605; Michael A. Proksch, Reg. No. 43,021; Thomas A. Jolly, Reg. No. 39,241; David A. Morasch, Reg. No. 42,905; Katie E. Sako, Reg. No. 32,628 and Daniel D. Crouse, Reg. No. 32,022.

Send correspondence to: LEE & HAYES, PLLC, 421 W. Riverside Avenue, Suite 500, Spokane, Washington, 99201. Direct telephone calls to: Allan T. Sponseller (509) 324-9256.

All statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statement may jeopardize the validity of the application or any patent issued therefrom.

* * * * *

Full name of inventor: Nosakhare D. Omoigui
Inventor's Signature  Date: 12/15/99
Residence: Redmond, WA
Citizenship: Nigeria
Post Office Address: 8935 160th Ave NE #A105
Redmond, WA 98052